

1 **In the Claims**

2 Please cancel claims 33-36 and 71-75 without prejudice.

3 Please amend claims 20-25, 27, 29, 37, 39, 41, 61, and 64 as shown herein.

4 Claims 1-32 and 37-70 are pending and are listed following:

5
6 1. **(original)** A network system, comprising:

7 a first device to maintain an original resource;

8 a second device to maintain a replica resource remotely from the first
9 device, the replica resource being replicated from the original resource;

10 memory to store a cached descriptor corresponding to the original resource;

11 a security component to determine whether the replica resource will pose a
12 security risk to the second device upon receipt of a request for the replica resource,
13 the security component:

14 formulating a descriptor corresponding to the replica resource and
15 comparing the formulated descriptor with the cached descriptor; and

16 if the formulated descriptor and the cached descriptor are not
17 equivalent, formulating a second descriptor corresponding to the original
18 resource and comparing the formulated descriptor with the second
19 descriptor.

20
21 2. **(original)** A network system as recited in claim 1, wherein the
22 security component determines that the replica resource is not a security risk if the
23 formulated descriptor and the cached descriptor are equivalent.

1 **3. (original)** A network system as recited in claim 1, wherein, if the
2 formulated descriptor and the cached descriptor are not equivalent, and if the
3 formulated descriptor and the second descriptor are equivalent, the security
4 component determines that the replica resource is not a security risk.

5
6 **4. (original)** A network system as recited in claim 1, wherein, if the
7 formulated descriptor and the cached descriptor are not equivalent, and if the
8 formulated descriptor and the second descriptor are equivalent, the security
9 component determines that the replica resource is not a security risk, and the
10 cached descriptor is replaced with the second descriptor.

11
12 **5. (original)** A network system as recited in claim 1, wherein, if the
13 formulated descriptor and the cached descriptor are not equivalent, and if the
14 formulated descriptor and the second descriptor are not equivalent, the security
15 component determines that the replica resource is a security risk, and the replica
16 resource is replaced with a copy of the original resource.

17
18 **6. (original)** A network system as recited in claim 1, wherein, if the
19 formulated descriptor and the cached descriptor are not equivalent, and if the
20 formulated descriptor and the second descriptor are not equivalent, the security
21 component determines that the replica resource is a security risk, the replica
22 resource is replaced with a copy of the original resource, and the cached descriptor
23 is replaced with the second descriptor.
24
25

1 7. **(original)** A network system as recited in claim 1, wherein the
2 security component formulates the cached descriptor when the original resource is
3 replicated to create the replica resource.

4
5 8. **(original)** A network system as recited in claim 1, wherein the
6 security component is configured to determine whether the request will pose a
7 security risk to the second device.

8
9 9. **(original)** A network system as recited in claim 8, wherein the
10 request designates a resource locator.

11
12 10. **(original)** A network system as recited in claim 8, wherein the
13 request designates a resource locator having a resource path, the resource path
14 identifying a location of the replica resource, and wherein the security component
15 determines that the request is not a security risk if the resource path does not
16 exceed a maximum number of characters.

17
18 11. **(original)** A network system as recited in claim 8, wherein the
19 request designates a resource locator having a plurality of arguments, and wherein
20 the security component determines that the request is not a security risk if
21 individual arguments do not exceed a maximum number of characters, and if a
22 total number of characters defining all of the arguments do not exceed a maximum
23 number of characters.

1 **12. (original)** A network system as recited in claim 8, wherein the
2 request designates a resource locator having a resource identifier, and wherein the
3 security component determines that the request is not a security risk if the resource
4 identifier has a valid file extension.

5
6 **13. (original)** A network system as recited in claim 1, wherein:
7 the request designates a resource locator having a resource path and one or
8 more arguments, the resource path identifying a location of the replica resource
9 and the resource path having a resource identifier;

10 the security component is configured to determine whether the request will
11 pose a security risk to the second device;

12 the security component determines that the request is not a security risk if:

13 the resource path does not exceed a maximum number of characters;

14 individual arguments do not exceed a maximum number of
15 characters;

16 a total number of characters defining all of the arguments do not
17 exceed a maximum number of characters; and

18 the resource identifier has a valid file extension.
19
20
21
22
23
24
25

1 **14. (original)** A network server, comprising:

2 a server component to receive a request for a resource maintained on the
3 network server and, in response to the request, implement security policies to
4 prevent unauthorized access to the resource; and

5 a security component that is registerable with the server component during
6 run-time to determine whether the request will pose a security risk to the network
7 server.

8
9 **15. (original)** A network server as recited in claim 14, wherein, if the
10 security component determines that the request will pose a security risk, the
11 security component redirects the request to indicate that the resource is not
12 available.

13
14 **16. (original)** A network server as recited in claim 14, wherein the
15 request designates a resource locator having a resource path, the resource path
16 identifying a location of the resource, and wherein the security component
17 determines that the request is not a security risk if the resource path does not
18 exceed a maximum number of characters.

1 **17. (original)** A network server as recited in claim 14, wherein the
2 request designates a resource locator having a plurality of arguments, and wherein
3 the security component determines that the request is not a security risk if
4 individual arguments do not exceed a maximum number of characters, and if a
5 total number of characters defining all of the arguments do not exceed a maximum
6 number of characters.

7
8 **18. (original)** A network server as recited in claim 14, wherein the
9 request designates a resource locator having a resource identifier, and wherein the
10 security component determines that the request is not a security risk if the resource
11 identifier has a valid file extension.

12
13 **19. (original)** A network server as recited in claim 14, wherein:
14 the request designates a resource locator having a resource path and one or
15 more arguments, the resource path identifying a location of the resource and the
16 resource path having a resource identifier;

17 the security component determines that the request is not a security risk if:

18 the resource path does not exceed a maximum number of characters;

19 individual arguments do not exceed a maximum number of
20 characters;

21 a total number of characters defining all of the arguments do not
22 exceed a maximum number of characters; and

23 the resource identifier has a valid file extension.
24
25

1 **20. (currently amended)** A network server system, comprising:

2 a server component in a network server to receive a request for a resource
3 maintained on the network server and, in response to the request, implement
4 security policies to prevent unauthorized access to the resource; and

5 a security component ~~that is~~ in a computing device remote to the network
6 server and registerable with the server component during run-time to determine
7 whether the resource will pose a security risk to the network server upon receipt of
8 the request.

9
10 **21. (currently amended)** A network server system as recited in
11 claim 20, wherein, if the security component determines that the resource will
12 pose a security risk, the security component redirects the request to indicate that
13 the resource is not available.

14
15 **22. (currently amended)** A network server system as recited in
16 claim 20, wherein the security component:

17 formulates a descriptor corresponding to the resource;

18 compares the formulated descriptor with a cached descriptor, the cached
19 descriptor corresponding to the resource and formulated when the resource is
20 initially requested; and

21 determines that the resource is not a security risk if the formulated
22 descriptor and the cached descriptor are equivalent.

1 **23. (currently amended)** A network server system as recited in
2 claim 20, wherein the security component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor; and

12 determines that the resource is not a security risk if the formulated
13 descriptor and the second descriptor are equivalent.
14
15
16
17
18
19
20
21
22
23
24
25

1 **24. (currently amended)** A network server system as recited in
2 claim 20, wherein the security component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent,
13 initiates that the resource stored on the network server be replaced with a copy of
14 the original resource maintained on the file server; and

15 initiates that the cached descriptor be replaced with the second descriptor.
16
17
18
19
20
21
22
23
24
25

1 **25. (currently amended)** A network server, comprising:

2 an Internet server to receive a request for a resource maintained on the
3 network server and, in response to the request, implement security policies to
4 prevent unauthorized access to the resource;

5 a security component that is registerable with the Internet server during
6 run-time, the security component having:

7 a validation component to determine whether the request will pose a
8 security risk to the network server by determining if a total number of
9 characters defining all of the arguments of the request exceeds a maximum
10 number of characters; and

11 an integrity verification component to determine whether the
12 resource will pose a security risk to the network server upon receipt of the
13 request.

14
15 **26. (original)** A network server as recited in claim 25, wherein the
16 request designates a resource locator having a resource path, the resource path
17 identifying a location of the resource, and wherein the validation component
18 determines that the request is not a security risk if the resource path does not
19 exceed a maximum number of characters.

1 **27. (currently amended)** A network server as recited in claim 25,
2 wherein the request designates a resource locator having a plurality of arguments,
3 and wherein the validation component determines that the request is not a security
4 risk if individual arguments do not exceed a maximum number of characters,~~and~~
5 ~~if a total number of characters defining all of the arguments do not exceed a~~
6 ~~maximum number of characters.~~

7
8 **28. (original)** A network server as recited in claim 25, wherein the
9 request designates a resource locator having a resource identifier, and wherein the
10 validation component determines that the request is not a security risk if the
11 resource identifier has a valid file extension.
12
13
14
15
16
17
18
19
20
21
22
23
24
25

1 **29. (currently amended)** A network server as recited in claim 25,
2 wherein:

3 the request designates a resource locator having a resource path and one or
4 more arguments, the resource path identifying a location of the resource and the
5 resource path having a resource identifier;

6 the validation component determines that the request is not a security risk
7 if:

8 the resource path does not exceed a maximum number of characters;

9 individual arguments do not exceed a maximum number of
10 characters; and

11 ~~a total number of characters defining all of the arguments do not~~
12 ~~exceed a maximum number of characters; and~~

13 the resource identifier has a valid file extension.
14

15 **30. (original)** A network server as recited in claim 25, wherein the
16 integrity verification component:

17 formulates a descriptor corresponding to the resource;

18 compares the formulated descriptor with a cached descriptor, the cached
19 descriptor corresponding to the resource and formulated when the resource is
20 initially requested; and

21 determines that the resource is not a security risk if the formulated
22 descriptor and the cached descriptor are equivalent.
23
24
25

1 **31. (original)** A network server as recited in claim 25, wherein the
2 integrity verification component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor; and

12 determines that the resource is not a security risk if the formulated
13 descriptor and the second descriptor are equivalent.

1 **32. (original)** A network server as recited in claim 25, wherein the
2 integrity verification component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent,
13 initiates that the resource stored on the network server be replaced with a copy of
14 the original resource maintained on the file server; and

15 initiates that the cached descriptor be replaced with the second descriptor.

16
17 **33-36. canceled**
18
19
20
21
22
23
24
25

1 **37. (currently amended)** One or more computer readable media
2 containing a security application, comprising:

3 a validation component to determine whether a request for a resource poses
4 a security risk by determining if a total number of characters defining all of the
5 arguments of the request exceeds a maximum number of characters; and

6 an integrity verification component to determine whether the resource poses
7 a security risk.

8
9 **38. (original)** Computer readable media as recited in claim 37,
10 wherein the request designates a resource locator having a resource path, the
11 resource path identifying a location of the resource, and wherein the validation
12 component determines that the request is not a security risk if the resource path
13 does not exceed a maximum number of characters.

14
15 **39. (currently amended)** Computer readable media as recited in
16 claim 37, wherein the request designates a resource locator having a plurality of
17 arguments, and wherein the validation component determines that the request is
18 not a security risk if individual arguments do not exceed a maximum number of
19 characters, ~~and if a total number of characters defining all of the arguments do not~~
20 ~~exceed a maximum number of characters.~~

1 **40. (original)** Computer readable media as recited in claim 37,
2 wherein the request designates a resource locator having a resource identifier, and
3 wherein the validation component determines that the request is not a security risk
4 if the resource identifier has a valid file extension.

5
6 **41. (currently amended)** Computer readable media as recited in
7 claim 37, wherein:

8 the request designates a resource locator having a resource path and one or
9 more arguments, the resource path identifying a location of the resource and the
10 resource path having a resource identifier;

11 the validation component determines that the request is not a security risk
12 if:

13 the resource path does not exceed a maximum number of characters;

14 individual arguments do not exceed a maximum number of
15 characters; and

16 ~~a total number of characters defining all of the arguments do not~~
17 ~~exceed a maximum number of characters; and~~

18 the resource identifier has a valid file extension.
19
20
21
22
23
24
25

1 **42. (original)** Computer readable media as recited in claim 37,
2 wherein the integrity verification component:

3 formulates a descriptor corresponding to the resource when the security
4 application receives the request;

5 compares the formulated descriptor with a cached descriptor, the cached
6 descriptor corresponding to the resource and formulated when the resource is
7 initially requested; and

8 determines that the resource is not a security risk if the formulated
9 descriptor and the cached descriptor are equivalent.

10
11 **43. (original)** Computer readable media as recited in claim 37,
12 wherein the integrity verification component:

13 formulates a descriptor corresponding to the resource when the security
14 application receives the request;

15 compares the formulated descriptor with a cached descriptor, the cached
16 descriptor corresponding to the resource and formulated when the resource is
17 initially requested;

18 if the formulated descriptor and the cached descriptor are not equivalent,
19 formulates a second descriptor corresponding to an original resource remotely
20 located, the resource being replicated from the original resource;

21 compares the formulated descriptor with the second descriptor; and

22 determines that the resource is not a security risk if the formulated
23 descriptor and the second descriptor are equivalent.

1 **44. (original)** Computer readable media as recited in claim 37,
2 wherein the integrity verification component:

3 formulates a descriptor corresponding to the resource when the security
4 application receives the request;

5 compares the formulated descriptor with a cached descriptor, the cached
6 descriptor corresponding to the resource and formulated when the resource is
7 initially requested;

8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulates a second descriptor corresponding to an original resource remotely
10 located, the resource being replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent,
13 initiates that the resource be replaced with a copy of the original resource; and

14 initiates that the cached descriptor be replaced with the second descriptor.
15
16
17
18
19
20
21
22
23
24
25

1 **45. (original)** A method, comprising:
2 receiving a request for a replica resource stored on a computing device;
3 formulating a descriptor corresponding to the replica resource;
4 comparing the formulated descriptor with a cached descriptor
5 corresponding to an original resource stored on a second computing device
6 remotely located from the computing device, the replica resource being replicated
7 from the original resource;
8 determining that the replica resource does not pose a security risk if the
9 formulated descriptor and the cached descriptor are equivalent;
10 if the formulated descriptor and the cached descriptor are not equivalent,
11 formulating a second descriptor corresponding to the original resource;
12 comparing the formulated descriptor with the second descriptor; and
13 determining that the replica resource does not pose a security risk if the
14 formulated descriptor and the second descriptor are equivalent.

15
16 **46. (original)** A method as recited in claim 45, further comprising
17 allowing the request if said determining that the replica resource does not pose a
18 security risk to the computing device.

19
20 **47. (original)** A method as recited in claim 45, further comprising
21 redirecting the request to indicate that the replica resource is not available if
22 determining that the replica resource poses a security risk to the computing device.
23
24
25

1 **48. (original)** A method as recited in claim 45, further comprising
2 replacing the cached descriptor with the second descriptor if the formulated
3 descriptor and the second descriptor are equivalent.

4
5 **49. (original)** A method as recited in claim 45, further comprising
6 replacing the replica resource with a copy of the original resource if the
7 formulated descriptor and the cached descriptor are not equivalent, and if the
8 formulated descriptor and the second descriptor are not equivalent.

9
10 **50. (original)** A method as recited in claim 45, further comprising
11 replacing the cached descriptor with the second descriptor if the formulated
12 descriptor and the cached descriptor are not equivalent, and if the formulated
13 descriptor and the second descriptor are not equivalent.

14
15 **51. (original)** A method as recited in claim 45, further comprising
16 formulating the cached descriptor when the original resource is replicated to create
17 the replica resource.

18
19 **52. (original)** A method as recited in claim 45, further comprising
20 formulating the cached descriptor when the replica resource is initially requested.

21
22 **53. (original)** A method as recited in claim 45, further comprising
23 determining whether the request will pose a security risk.
24
25

1 **54. (original)** A method as recited in claim 45, further comprising:
2 determining whether the request will pose a security risk; and
3 redirecting the request to indicate that the replica resource is not available if
4 determining that the request poses a security risk to the computing device.

5
6 **55. (original)** A method as recited in claim 45, wherein the request
7 designates a resource locator having a resource path, the resource path identifying
8 a location of the replica resource, and the method further comprising determining
9 that the request does not pose a security risk if the resource path does not exceed a
10 maximum number of characters.

11
12 **56. (original)** A method as recited in claim 45, wherein the request
13 designates a resource locator having a plurality of arguments, and the method
14 further comprising determining that the request does not pose a security risk if
15 individual arguments do not exceed a maximum number of characters, and if a
16 total number of characters defining all of the arguments do not exceed a maximum
17 number of characters.

18
19 **57. (original)** A method as recited in claim 45, wherein the request
20 designates a resource locator having a resource identifier, and the method further
21 comprising determining that the request does not pose a security risk if the
22 resource identifier has a valid file extension.

1 **58. (original)** A method as recited in claim 45, wherein:

2 the request designates a resource locator having a resource path and one or
3 more arguments, the resource path identifying a location of the replica resource
4 and the resource path having a resource identifier;

5 the method further comprising determining that the request does not pose a
6 security risk if:

7 the resource path does not exceed a maximum number of characters;

8 individual arguments do not exceed a maximum number of
9 characters;

10 a total number of characters defining all of the arguments do not
11 exceed a maximum number of characters; and

12 the resource identifier has a valid file extension.

13
14 **59. (original)** A computer-readable medium comprising computer
15 executable instructions that, when executed, direct a computing system to perform
16 the method of claim 45.

17
18 **60. (original)** A computer-readable medium comprising computer
19 executable instructions that, when executed, direct a computing system to perform
20 the method of claim 58.

1 **61. (currently amended)** A method, comprising:

2 receiving a request for a resource;

3 implementing security policies to prevent unauthorized access to the
4 resource;

5 determining whether the request will pose a security risk by determining if
6 a total number of characters defining all of the arguments of the request exceeds a
7 maximum number of characters; and

8 determining whether the resource will pose a security risk if allowing the
9 request.

10
11 **62. (original)** A method as recited in claim 61, further comprising
12 allowing the request for the resource if determining that the request does not pose
13 a security risk and if determining that the resource does not pose a security risk.

14
15 **63. (original)** A method as recited in claim 61, wherein the request
16 designates a resource locator having a resource path, the resource path identifying
17 a location of the resource, and the method further comprising determining that the
18 request does not pose a security risk if the resource path does not exceed a
19 maximum number of characters.

1 **64. (currently amended)** A method as recited in claim 61, wherein
2 the request designates a resource locator having a plurality of arguments, and the
3 method further comprising determining that the request does not pose a security
4 risk if individual arguments do not exceed a maximum number of characters, ~~and~~
5 ~~if a total number of characters defining all of the arguments do not exceed a~~
6 ~~maximum number of characters.~~

7
8 **65. (original)** A method as recited in claim 61, wherein the request
9 designates a resource locator having a resource identifier, and the method further
10 comprising determining that the request does not pose a security risk if the
11 resource identifier has a valid file extension.

12
13 **66. (original)** A method as recited in claim 61, further comprising:
14 formulating a descriptor corresponding to the resource;
15 comparing the formulated descriptor with a cached descriptor
16 corresponding to the resource and formulated when the resource is initially
17 requested; and
18 determining that the resource does not pose a security risk if the formulated
19 descriptor and the cached descriptor are equivalent.
20
21
22
23
24
25

1 **67. (original)** A method as recited in claim 61, further comprising:
2 formulating a descriptor corresponding to the resource;
3 comparing the formulated descriptor with a cached descriptor
4 corresponding to the resource and formulated when the resource is initially
5 requested;
6 determining that the resource does not pose a security risk if the formulated
7 descriptor and the cached descriptor are equivalent;
8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulating a second descriptor corresponding to an original resource remotely
10 located, the resource replicated from the original source;
11 comparing the formulated descriptor with the second descriptor; and
12 determining that the resource does not pose a security risk if the formulated
13 descriptor and the second descriptor are equivalent.

1 **68. (original)** A method as recited in claim 61, further comprising:
2 formulating a descriptor corresponding to the resource;
3 comparing the formulated descriptor with a cached descriptor
4 corresponding to the resource and formulated when the resource is initially
5 requested;
6 determining that the resource does not pose a security risk if the formulated
7 descriptor and the cached descriptor are equivalent;
8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulating a second descriptor corresponding to an original resource remotely
10 located, the resource replicated from the original resource;
11 comparing the formulated descriptor with the second descriptor; and
12 determining that the resource does not pose a security risk if the formulated
13 descriptor and the second descriptor are equivalent;
14 if the formulated descriptor and the second descriptor are not equivalent,
15 replacing the resource with a copy of the original resource and replacing the
16 cached descriptor with the second descriptor.

17
18 **69. (original)** A computer-readable medium comprising computer
19 executable instructions that, when executed, direct a computing system to perform
20 the method of claim 61.

21
22 **70. (original)** A computer-readable medium comprising computer
23 executable instructions that, when executed, direct a computing system to perform
24 the method of claim 68.
25

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

71-75. canceled

This Page Blank (uspto)